A Modified Iterative Water-filling Algorithm with Per-iteration Power Normalization in Multiuser MMSE-Precoded MIMO Systems

Min Lee and Seong Keun Oh
School of Electrical and Computer Engineering
Ajou University, Korea
E-mails: {minishow, oskn}@ajou.ac.kr

Abstract—In this paper, we deal with an optimum per-user power allocation problem in terms of the sum capacity in multiuser multiple input multiple output (MU-MIMO) downlink systems when the minimum mean square error (MMSE) scheme has been used for multiuser precoding. However, MMSE precoding allows some inter-user interference. In this paper, we propose a modified iterative water-filling (MIWF) algorithm with per-iteration power normalization (PIPN) in multiuser MMSE-precoded MIMO systems. In the proposed algorithm, power levels for respective users are re-adjusted prior to the consecutive iteration, so that the total transmit power constraint should be satisfied. This normalization can reduce greatly the number of iterations for the iterative water-filling process since the inter-user interference terms at every iteration can be computed more accurately without performance degradation. From computer simulations and complexity analyses, we see that the proposed algorithm has lower complexity but with the same capacity as compared with the original MIWF algorithm.

Keywords—Multiuser MIMO; MMSE precoding; Modified iterative water-filling

I. INTRODUCTION

During the last decade, MIMO techniques have drawn a lot of attentions, because of their promising improvement in terms of both spectral efficiency and performance [2], [3]. Among many MIMO issues, MU-MIMO precoding is currently one of hottest topics, since the MU-MIMO precoding exploits synergically the high capacity achievable with MIMO multiplexing and the benefit of space division multiple access (SDMA). It also makes mobile terminals simple, since a lot of complex signal processing tasks are done in advance at the transmitter [4].

So far, various MU-MIMO precoding techniques including zero-forcing (ZF) technique, block diagonalization (BD) technique [5], successive MMSE (SMMSE) techniques [6]-[7], Tomlinson- Harashima precoding (THP) techniques [8], successive optimization THP (SO-THP) technique [9] have been developed. Among them, the MMSE precoding technique [10] has not only the benefits of SDMA, but also provides a reasonable performance when all user terminals are equipped with a single receive antenna. The technique has relatively low computational complexity due to linear transmit processing as compared with non-linear precoding techniques. Compared with the ZF technique that enhances the noise, the MMSE technique improves the system performance due to minimizing composite interference-plus-noise power through allowing a certain amount of inter-stream interference especially at low SNR.

Proper Power allocation can result in significant performance improvement. However, the problem of optimizing per-user power allocation under inter-stream interference is difficult to solve, since it is not convex. Recently, the MIWF was proposed for per-user power allocation to maximize the sum capacity under inter-stream interference [1]. The algorithm is capable of finding sub-optimal solutions due to a taxation scheme that takes into account the interacting effect of inter-stream interference in an iterative manner. However, in the MIWF algorithm, per-user power levels at every iteration for the inner loop cannot be satisfied the total power constraint. Hence, it may result in inaccurate inter-user interference terms and can delay the iterative process to converge to the solutions.

In this paper, we propose a MIWF algorithm with PIPN for per-user power allocation in multiuser MMSE-precoded MIMO downlink systems, to maximize the sum capacity. In the proposed algorithm, per-user power levels at every iteration for the inner loop are normalized so that the total transmit power constraint could be satisfied, prior to being applied for consecutive iteration. This normalization can make that the inter-user interference terms are computed accurately at every iteration and reduce the number of iterations for the iterative water-filling process, as compared with the original MIWF algorithm, without any performance loss.

We present the system model of the MU-MIMO downlink system with precoding in Section 2, and introduce the MIWF algorithm with PIPN in Section 3. After presenting the simulation results in Section 4, we analyze the complexity in Section 5.

II. SYSTEM MODEL

We consider a Gaussian MIMO broadcast (BC) channel with a single transmitter communicating independent information to $K$ users. The block diagram of MU-MIMO downlink system is illustrated in Fig. 1. In what follows, we assume channel state information (CSI) knowledge at both the base station (BS) and mobile stations (MSs) for downlink precoding. $M$ and $K$ are used to represent the number of transmit antennas and receive antennas, i.e. there are $K$ users quipped with a single antenna.
Fig. 1. MU-MIMO downlink system with precoding.

The received signal vector at MSs is

\[ r = H F p + n , \]  

(1)

where \( r = (r_1, r_2, \ldots, r_K)^T \) is a column vector consist of the received signals at MSs. \( H \) is a \( K \times M \) channel matrix. \( F \) is a \( M \times K \) precoding matrix, \( p \) is a \( K \times K \) diagonal matrix for power allocation, \( s \) is an \( 1 \times K \) column vector for transmit signals, and \( n \) is a \( 1 \times K \) column vector for zero-mean additive Gaussian noises at MSs. Here, \( H \) is the MU-MIMO BC channel matrix made up of per-user row channel vectors as follows:

\[ H = \begin{bmatrix} h_1 \\ \vdots \\ h_K \end{bmatrix}, \]  

(2)

where \( h_i (i=1,2,\ldots,K) \) denotes an \( 1 \times M \) per-user MISO channel vector made up of channel gains from transmit antennas to receive antenna for the \( i \)-th user. The precoding matrix \( F \) again consists of combining per-user precoding vectors as

\[ f = [f_1 \ \cdots \ f_K], \]  

(3)

where \( f \) denotes a \( M \times 1 \) column vector of per-user precoding vector for the \( i \)-th user. The power allocation matrix is made up of square root power allocated users’ signals as

\[ p = \begin{bmatrix} \sqrt{p_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sqrt{p_K} \end{bmatrix}, \]  

(4)

where \( \sqrt{p_i} \) is a square root power allocated the signal of \( i \)-th user. The transmitted signal vector \( s \) can also be decomposed as

\[ s = [s_1 \ \cdots \ s_K]^T, \]  

(5)

where \( s_i \) is a signal transmitted to the \( i \)-th user. The noise vector \( n \) is decomposed as

\[ n = [n_1 \ \cdots \ n_K]^T, \]  

(6)

where \( n_i \) is an additive white Gaussian noise at the mobile receiver of the \( i \)-th user.

From (2) through (6), we can rewrite (1) as

\[ \begin{bmatrix} r_1 \\ \vdots \\ r_K \\ \sqrt{p_i} \end{bmatrix} = \begin{bmatrix} h_1 \\ \vdots \\ h_k \\ \sqrt{p_i} \end{bmatrix} \begin{bmatrix} f_1 \\ \vdots \\ f_k \\ 0 \end{bmatrix} + \begin{bmatrix} n_1 \\ \vdots \\ n_k \\ \sqrt{p_i} \end{bmatrix}, \]  

(7)

Hence, the received signal of the \( i \)-th user can then be expressed as

\[ r_i = h_i \sum_{k=1}^{K} f_k \sqrt{p_k} s_k + n_i, \]  

\[ i = 1, 2, \ldots, K. \]  

(8)

III. MODIFIED ITERATIVE WATER-FILLING ALGORITHM WITH PER-ITERATION POWER NORMALIZATION

In this section, we present the MIWF algorithm with PIPN for multiuser MMSE-precoded MIMO downlink systems with inter-user interference. To find an effective algorithm to solve the problem of optimizing per-user power allocation under inter-stream interference, we invoke some iterative approaches to solve the Karush-Kuhn-Tucker (KKT) system of the non-convex optimization problem.

When the inter-stream interference is regarded as noise, the \( K \)-user optimization problem to maximize the sum capacity can be expressed as

\[
\max \sum_{i=1}^{K} \log_2 \left( 1 + \frac{p_i |\tilde{h}_{ij}|^2}{\sigma_n^2 + \sum_{j \neq i} p_j |\tilde{h}_{ij}|^2} \right) \\
\text{s.t.} \sum_{i=1}^{K} p_i \leq P_T, \quad p_i \geq 0 \quad i = 1, \ldots, K,
\]  

(9)

where \( \tilde{h}_{ij} \) is an effective channel gain experienced by the signal of \( j \)-th user that interferes with the \( i \)-th user, and \( P_T \) denotes the total power constraint. Then, the effective channel matrix can be expressed as

\[ H_{\text{eff}} = H \cdot F = \begin{bmatrix} \tilde{h}_{11} & \tilde{h}_{12} & \cdots & \tilde{h}_{1K} \\ \tilde{h}_{21} & \tilde{h}_{22} & \cdots & \tilde{h}_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{h}_{K1} & \tilde{h}_{K2} & \cdots & \tilde{h}_{KK} \end{bmatrix}. \]  

(10)

Consider the Lagrangian of the optimization problem (9), we can write down a following solution using Lagrangian multiplier:

\[ L \left( \{ p_i \}, \mu \right) = \sum_{i=1}^{K} \log_2 \left( 1 + \frac{p_i |\tilde{h}_{ij}|^2}{\sigma_n^2 + \sum_{j \neq i} p_j |\tilde{h}_{ij}|^2} \right) - \mu \left( \sum_{i=1}^{K} p_i - P_T \right). \]  

(11)

By taking the derivative of the above with respect to \( p_i \), we can obtain the KKT system of the optimization problem:

\[ \frac{1}{|\tilde{h}_{ij}|^2} \frac{1}{p_i + \sum_{j \neq i} p_j |\tilde{h}_{ij}|^2} = \mu, \]  

(12)

where \( \mu \) defined as follows:
\[ t_i = \frac{\sum_{j=\{h_{ij}\}}^{K} p_j \left| \tilde{h}_{ij} \right|^2}{\sum_{j=\{h_{ij}\}}^{K} p_j \left| \tilde{h}_{ij} \right|^2 + \sum_{k=\{K+1\}}^{K} p_k \left| \tilde{h}_{ik} \right|^2 + \sigma_n^2} \times \left( \frac{\left| \tilde{h}_{ij} \right|^2}{\sum_{k=\{K+1\}}^{K} p_k \left| \tilde{h}_{ik} \right|^2 + \sigma_n^2} \right). \tag{13} \]

Solving the optimization problem (9) can now be thought of as solving the KKT system (12)-(13) along with the power constraint and positivity constraints on \( p_i \) and the equality in (12) needs to be replaced by less than or equal to, when \( p_i = 0 \).

The \( t_i \) term summarizes the effect of interference that \( i \)-th user causes to other users and can be interpreted as a taxation term. To solve this nonconvex optimization problem, we fix both the \( t_i \) and the interference term from other users, denoted by \( I_i \), where

\[ I_i = \sum_{j=\{h_{ij}\}}^{K} p_j \left| \tilde{h}_{ij} \right|^2, \tag{14} \]

and solve for using (12). We then update the \( t_i \) and \( I_i \) terms according to the newly obtained \( p_i \) and repeat the process until convergence.

This strategy is numerically efficient because (12) is essentially a modified water-filling step, which can be rewritten as:

\[ p_i + \frac{I_i}{\left| \tilde{h}_{ij} \right|^2} + \frac{\sigma_n^2}{\left| \tilde{h}_{ij} \right|^2} = \frac{1}{\mu + t_i}. \tag{15} \]

Fixing \( t_i \) and \( I_i \), iterative water-filling algorithm finds the optimal \( p_i \). The first step is to recognize that given \( \mu, p_i \) can be obtained from (15) as follows:

\[ p_i = \left( \frac{1}{\mu + t_i} - \frac{1}{\left| \tilde{h}_{ij} \right|^2} + \frac{\sigma_n^2}{\left| \tilde{h}_{ij} \right|^2} \right)^+. \tag{16} \]

where \( x^+ := \max(x, 0) \). Here, \( \mu \) is essentially the inverse of the water-filling level but modified by \( t_i \). Next to find \( \mu \), we can sum (16) over \( i \) and get

\[ P_T = \sum_{i=1}^{K} \left( \frac{1}{\mu + t_i} - \frac{I_i}{\left| \tilde{h}_{ij} \right|^2} + \frac{\sigma_n^2}{\left| \tilde{h}_{ij} \right|^2} \right)^+. \tag{17} \]

With \( t_i \) and \( I_i \) fixed, this is now an equation of a single variable \( \mu \). Further, the right-hand side of (17) is a monotonic function of \( \mu \). Thus, (17) can be solved efficiently via a one dimensional search (e.g. using bisection). After \( \mu \) is found, \( p_i \) can then be obtained from (16).

If zero power is assigned to some users in (16), an additional power allocation process should be done with exemption of zero-powered users. However, in the MIWF algorithm, the interference terms in (14) should be re-computed if the number of users is changed. In this case, the total power constraint should be satisfied prior to being applied for consecutive iteration.

Once the optimum \( \{ p_i, \mu \} \) is obtained, we may then operate water-filling loop over the total users, taking into account the updated interference terms at every iteration for the inner loop of iterative water-filling until the process converges. Finally, in an outer loop, we may then update \( t_i \) according to (13) and repeat the process until the entire KKT system is solved. The algorithm is summarized in the following:

**Algorithm** Consider a \( K \)-user system. Assume the total power constraint \( P_T \). The power allocation algorithm works as follows:

1. Initialize \( p_i^{(0)} = P_T / K \), \( n = 0 \)
2. Repeat (outer loop)
   - \( n = n + 1 \)
   - Repeat (inner loop)
     - Fixing \( t_i \) and \( I_i \), iterative water-filling algorithm finds the optimal \( p_i \). The first step is to recognize that given \( \mu, p_i \) can be obtained from (15) as follows:
     \[ p_i = \left( \frac{1}{\mu + t_i} - \frac{I_i}{\left| \tilde{h}_{ij} \right|^2} + \frac{\sigma_n^2}{\left| \tilde{h}_{ij} \right|^2} \right)^+. \]
     - Obtain \( \mu \) via bisection on \( P_T = \sum_{i=1}^{K} \left( \frac{1}{\mu + t_i} - \frac{I_i}{\left| \tilde{h}_{ij} \right|^2} + \frac{\sigma_n^2}{\left| \tilde{h}_{ij} \right|^2} \right) \)
     - Normalize to satisfy \( P_T = \sum_{i=1}^{K} p_i \)
   - Until \( \mu \) converges.
     - Fixing \( t_i \) and \( I_i \), iterative water-filling algorithm finds the optimal \( p_i \). The first step is to recognize that given \( \mu, p_i \) can be obtained from (15) as follows:
     \[ p_i = \left( \frac{1}{\mu + t_i} - \frac{I_i}{\left| \tilde{h}_{ij} \right|^2} + \frac{\sigma_n^2}{\left| \tilde{h}_{ij} \right|^2} \right)^+. \]
     - Obtain \( \mu \) via bisection on \( P_T = \sum_{i=1}^{K} \left( \frac{1}{\mu + t_i} - \frac{I_i}{\left| \tilde{h}_{ij} \right|^2} + \frac{\sigma_n^2}{\left| \tilde{h}_{ij} \right|^2} \right) \)
     - Normalize to satisfy \( P_T = \sum_{i=1}^{K} p_i \)
     - Until \( \mu \) converges.
   - Repeat (outer loop)
     - \( n = n + 1 \)
     - Repeat (inner loop)
   - Until \( \mu \) converges.

The idea of using the term \( t_i \) to take into account the effect of interference each user causes onto others has been used in the past (e.g. [11],[12]). We can find the water-filling level \( \mu \) efficiently using bisection due to the incorporation of the \( t_i \) term in the water-filling process. The \( t_i \) term lowers the water-filling level where the effect of its interference to other users is strong.

**IV. SIMULATION RESULTS**

We performed computer simulations to evaluate the performance of the proposed algorithm and to compare it with that of the original MIWF algorithm [1] for MU-MIMO downlink system with MMSE precoding. The channel \( H \) was assumed to be spatially white and flat Rayleigh fading. We use the notation \( \{ M_K, \ldots, M_K \} \times M_T \) to describe the antenna configuration of the MU-MIMO systems.
In the simulations for ergodic sum capacity, we considered $\{1,1,1,1\} \times 4$ configuration. The receive signal-to-noise ratio defined as $\text{SNR}_r = \frac{P_t}{\sigma_n^2}$ was used [9].

Fig. 2 shows ergodic sum capacity of the MIWF algorithm with PIPN and the original MIWF algorithm. From Fig. 2, we see that both algorithms have equal performance, since both algorithms use the iterative water-filling algorithms using taxation scheme to solve optimization problem.

![Fig. 2. Sum capacity of the MIWF algorithm with PIPN and the original MIWF algorithm.](image)

In this section, we would compare the proposed power allocation algorithm with the original MIWF algorithm in terms of computational complexity.

For simplicity, we assume that the number of transmit antenna is equal to the number of users equipped with a single receive antenna, as $K$. In the following operation counts for computational complexity analysis, we take account of the number of iterations for the outer loop of iterative water-filling, since the computational complexity of iterative water-filling depends on the number of iterations of the outer loop. We also take account of the number of multiplications because the operation count based on multiplications alone is usually a good measure of relative efficiency.

In counting the number of multiplications, we consider the multiplications with respect to bisection for finding the water-filling level, normalization for only the proposed algorithm, and update the taxation and interference terms. This is why these four terms reflect the differences between the proposed and the original MIWF algorithms.

Let $\tilde{K}$ is the number of users who are allocated power more than zero for the inner loop of the iterative water-filling. Bi-section method finds the water-filling level through iterative way. It requires $4\tilde{K} + 1$ multiplications at every iteration for the inner loop. The taxation and interference terms, respectively, require $3K^2 - K$, $K^2 - K$ multiplications from (13) and (14) at every iteration. We summarize the number of multiplications required for bisection, normalization, interference, and taxation at every iteration in the Table 1.

<table>
<thead>
<tr>
<th>Function</th>
<th>Number of multiplications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bisection</td>
<td>$4\tilde{K} + 1$</td>
</tr>
<tr>
<td>Normalize</td>
<td>$\tilde{K}$</td>
</tr>
<tr>
<td>Taxation</td>
<td>$3K^2 - K$</td>
</tr>
<tr>
<td>Interference</td>
<td>$K^2 - K$</td>
</tr>
</tbody>
</table>

We analyze the computational complexity with the total number of multiplications which are required above four terms and the number of iterations of the outer loop, inner loop, and bisection loop. We count the number of the number of multiplications required for 10000 channel realizations in the MIWF algorithm with PIPN and the original MIWF algorithm, respectively. Then we get the average number of multiplications by dividing the number of channel realizations into it.

Fig. 3 shows the average number of iterations for the outer loop in the MIWF algorithm with PIPN and the original MIWF algorithm according to the number of transmit antennas (and users). From Fig. 3, we can see that the proposed algorithm requires about 7% lower iterations of the outer loop than the original MIWF algorithm, excluding when the number of transmit antennas (and users) is 2.

Fig. 4 shows the average number of multiplications for obtaining the solutions in the MIWF algorithm with PIPN and the original MIWF algorithm according to the number of transmit antennas (and users). From Fig. 4, we can see that the proposed algorithm requires fewer multiplications than the original MIWF algorithm. We can see that the proposed algorithm requires about 11% lower multiplications than the original MIWF algorithm.

![Table 1. Number of multiplications for calculations required iterative water-filling algorithm](image)

<table>
<thead>
<tr>
<th>Function</th>
<th>Number of multiplications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bisection</td>
<td>$4\tilde{K} + 1$</td>
</tr>
<tr>
<td>Normalize</td>
<td>$\tilde{K}$</td>
</tr>
<tr>
<td>Taxation</td>
<td>$3K^2 - K$</td>
</tr>
<tr>
<td>Interference</td>
<td>$K^2 - K$</td>
</tr>
</tbody>
</table>

In this paper, we proposed the MIWF algorithm with PIPN for per-user power allocation suitable for the use in multiuser MMSE-precoded MIMO downlink systems, to maximize the sum capacity. In the proposed algorithm, per-user power levels at every iteration for the inner loop are normalized so that the total transmit power constraint should be satisfied, prior to being applied for the consecutive iteration. This can reduce the number of iterations for the iterative process since the inter-user interference terms at every iteration can be computed more accurately, as compared with the original MIWF algorithm, without any performance degradation. From simulation results and complexity analyses, the proposed algorithm has lower complexity although the identical capacity, as compared with the original MIWF algorithm.

VI. CONCLUSIONS

In this paper, we proposed the MIWF algorithm with PIPN for per-user power allocation suitable for the use in multiuser MMSE-precoded MIMO downlink systems, to maximize the sum capacity. In the proposed algorithm, per-user power levels at every iteration for the inner loop are normalized so that the total transmit power constraint should be satisfied, prior to being applied for the consecutive iteration. This can reduce the number of iterations for the iterative process since the inter-user interference terms at every iteration can be computed more accurately, as compared with the original MIWF algorithm, without any performance degradation. From simulation results and complexity analyses, the proposed algorithm has lower complexity although the identical capacity, as compared with the original MIWF algorithm.
ACKNOWLEDGEMENT

"This research was supported by the MIC(Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Advancement)" (IITA-2008-(C1090-0801-0003))

REFERENCES


